Autonomously Generating Operations Sequences For a Mars Rover using AI-based Planning

Rob Sherwood, Andrew Mishkin, Tara Estlin, Steve Chien, Paul Backes, Brian Cooper, Scott Maxwell, Gregg Rabideau

Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Dr.
Pasadena, CA 91109
818-393-5378
firstname.lastname@jpl.nasa.gov

Abstract

This paper discusses a proof-of-concept prototype for ground-based automatic generation of validated rover command sequences from high-level science and engineering activities. This prototype is based on ASPEN, the Automated Scheduling and Planning Environment. This Artificial Intelligence (AI) based planning and scheduling system will automatically generate a command sequence that will execute within resource constraints and satisfy flight rules. An automated planning and scheduling system encodes rover design knowledge and uses search and reasoning techniques to automatically generate low-level command sequences while respecting rover operability constraints, science and engineering preferences, environmental predictions, and also adhering to hard temporal constraints. This prototype planning system has been field-tested using the Rocky-7 rover at JPL, and will be field-tested on more complex rovers to prove its effectiveness before transferring the technology to flight operations for an upcoming NASA mission. Enabling goal-driven commanding of planetary rovers greatly reduces the requirements for highly skilled rover engineering personnel. This in turn greatly reduces mission operations costs. In addition, goal-driven commanding permits a faster response to changes in rover state (e.g., faults) or science discoveries by removing the time consuming manual sequence validation process, allowing rapid "what-if" analyses, and thus reducing overall cycle times.

Introduction

Unlike more traditional deep space missions, surface roving missions must be operated in a reactive mode, with mission planners waiting for an end of day telemetry downlink--including critical image data--in order to plan the next day's worth of activities. Communication time delays over interplanetary distances preclude simple 'joysticking' of the rover. A consequence of this approach to operations is that the full cycle of telemetry receipt, science and engineering analysis, science plan generation, command sequence generation and validation, and uplink of the sequence, must typically be performed in twelve hours or less. Yet current rover sequence generation is manual (Mishkin, et al., 1998), with limited ability to automatically generate valid rover activity sequences from more general activities/goals input by science and engineering team members. Tools such as the Rover Control Workstation (RCW) and the Web Interface for Telescience (WITS) provide mechanisms for human operators to manually generate plans and command sequences. (Backes, et. al, 1998) These tools even estimate some types of resource usage and identify certain flight rule violations. However, they do not provide any means to modify the plan in response to the constraints imposed by available resources or flight rules, except by continued manual editing of sequences. This current situation has two drawbacks. First, the operator-intensive construction and validation of sequences puts a tremendous workload on the rover engineering team. The manual process is error-prone, and can lead to operator fatigue over the many months of mission operations. Second, the hours that must be reserved for sequence generation and validation reduces the time available to the science team to identify science targets and formulate a plan for submission to the engineering team. This results in reduced science return. An automated planning tool would allow the science team and sequence team to work together to optimize the plan. Many different plan options could be explored. The faster turnaround of automated planning also permits shorter than once a day planning cycles.

The RCW software, used to operate the Sojourner rover during the Pathfinder mission, provides visualization for vehicle traverse (movement) planning, a command interface, constraint checking for individual commands, and some resource estimation (for sequence execution time and telemetry volume). However, this tool was never intended for automated goal-based planning of rover activities. To deal with these issues, there is a need for a new tool that is specifically geared toward automated planning.

We are using AI planning/scheduling technology to automatically generate valid rover command sequences from activity sequences specified by the mission science and engineering team. This system will automatically generate a command sequence that will execute within resource constraints and satisfy flight rules. Commanding the rover to achieve mission goals requires significant knowledge of the rover design, access to the low-level rover command set, and an understanding of the performance metrics rating the desirability of alternative sequences. It also requires coordination with external events such as orbiter passes and day/night cycles. An automated planning and scheduling system encodes this knowledge and uses search and reasoning techniques to automatically generate low-level command sequences while respecting rover operability constraints, science and engineering preferences, and also adhering to hard temporal constraints. A ground-based interactive planner combines the power of automated reasoning and conflict resolution techniques with the insights of the Science Team or Principal Investigator (PI) to prioritize and re-prioritize mission goals.

ASPEN Planning System

Planning and scheduling technology offers considerable promise in automating rover operations. Planning and scheduling rover operations involves generating a sequence of low-level commands from a set of high-level science and engineering goals.

ASPEN (Chien, et al., 2000) is an object-oriented planning and scheduling system that provides a reusable set of software components that can be tailored to specific domains. These components include:

- An expressive constraint modeling language to allow the user to define naturally the application domain
- A constraint management system for representing and maintaining spacecraft and rover operability and resource constraints, as well as activity requirements
- A set of search strategies for plan generation and repair to satisfy hard constraints
- A language for representing plan preferences and optimizing these preferences
- A soft, real-time replanning capability
- A temporal reasoning system for expressing and maintaining temporal constraints
- A graphical interface for visualizing plans/schedules (for use in mixed-initiative systems in which the problem solving process is interactive).

In ASPEN, the main algorithm for automated planning and scheduling is based on a technique called *iterative repair* (Rabideau, et al., 1999, Zweben et al., 1994). During iterative repair, the conflicts in the schedule are detected and addressed one at a time until conflicts no longer exist, or a user-defined time limit has been exceeded. A conflict is a violation of a resource limitation, parameter dependency or temporal constraint. Conflicts can be repaired by means of several predefined methods. The repair methods are: moving an activity, adding a new instance of an activity, deleting an activity, detailing an activity, abstracting an activity, making a resource reservation of an activity, canceling a reservation, connecting a temporal constraint, disconnecting a constraint, and changing a parameter value. The repair algorithm may use any of these methods in an attempt to resolve a conflict. How the algorithm performs is largely dependent on the type of conflict being resolved.

Rover knowledge is encoded in ASPEN under seven core model classes: activities, parameters, parameter dependencies, temporal constraints, reservations, resources and state variables. An activity is an occurrence over a time interval that in some way affects the rover. It can represent anything from a high-level goal or request to a low-level event or command. Activities are the central structures in ASPEN, and also the most complicated. Together, these constructs can be used to define rover procedures, rules and constraints in order to allow manual or automatic generation of valid sequences of activities, also called plans or schedules.

Once the types of activities are defined, specific instances can be created from the types. Multiple activity instances created from the same type might have different parameter values, including the start time. Many camera-imaging activities, for example, can be created from the same type but with different image targets and at different start times. The sequence of activity instances is what defines the plan.

The flight rules and constraints are defined within the activities. The flight rules can be defined as temporal constraints, resource constraints, or system state constraints. Temporal constraints are defined between activities. An example would be that the rate sensor must warm up for two to three minutes before a rover traverse. In ASPEN, this would be modeled within the "move rover" activity as shown in Figure 1. The rate_sensor_heat_up is another activity that is presumed to turn on a rate sensor heater.

```
Activity move_rover {
constraints =
starts after end_of rate_sensor_heat_up by [2m,3m];
reservations =
solar_array_power use 35,
rate_sensor_state change_to "on",
target_state must_be "ready";
};
```

Figure 1 - ASPEN Modeling Language Example

Constraints can also be state or resource related. State constraints can either require a particular state or change to a particular state. Resource constraints can use a particular amount of a resource. Resources with a capacity of one are called atomic resources. ASPEN also uses non-depletable and depletable resources. Non-depletable resources are resources that can used by more than one activity at a time and do not need to be replenished. Each activity can use a different quantity of the resource. An example would be the rover solar array power. Depletable resources are similar to non-depletable except that their capacity is diminished after use. In some cases their capacity can be replenished (memory capacity) and in other cases it cannot (battery energy, i.e. non-rechargeable primary batteries). Resource and state constraints are defined within activities using the keyword "reservations." See Figure 1 for an example.

The job of a planner/scheduler, whether manual or automated, is to accept high-level goals and generate a set of low-level activities that satisfy the goals and do not violate any of the rover flight rules or constraints. Goal-based rover planning requires significant knowledge of the rover design, access to the low-level rover command set, and an understanding of the performance metrics rating the desirability of alternative sequences. It also requires coordination with external events such as orbiter passes and day/night cycles. ASPEN provides a Graphical User Interface (GUI) for manual generation and/or manipulation of activity sequences. Figure 2 contains a screen dump of the GUI.

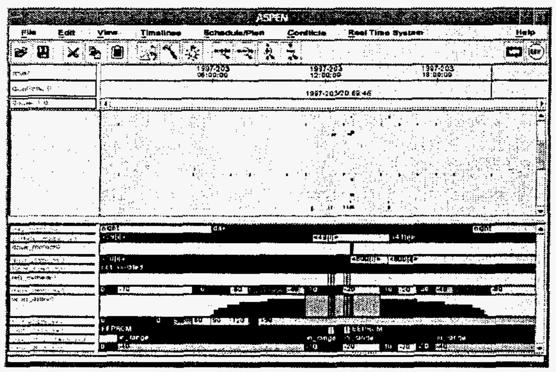

Figure 2 - ASPEN GUI

Status

Initial work in 1998 consisted of a preliminary proof of concept demonstration in which we used automated planning and scheduling technology integrated with WITS to demonstrate automated commanding for the Rocky-7 rover from the WITS interface. (Backes, et al., 1999) The Rocky7 research rover has been developed at JPL by the Long Range Science Rover task of the NASA Telerobotics program. The rover was placed in the JPL Mars Yard, a simulated Mars landscape. Next, a set of panorama images was taken by the 1.5 meter tall deployable mast on the rover. Using these images, the WITS user selected a dig target locations, science imaging targets, spectrometer imaging targets, and their associated parameters and priorities. The WITS tool was used to visualize the terrain around the rover, generate the initial science targets and activities, and to send the final sequence to the Rocky7 rover. ASPEN utilized automated resource analysis, planning, and scheduling to take the initial sequence from WITS and generate a more complete and valid final sequence, which was returned to WITS. The final sequence was then executed on the rover in the JPL Mars Yard.

The focus of our recent work has been to compare the automated ground-based commanding tool to the manual commanding process of the Mars Pathfinder Sojourner rover. The engineering model of the Sojourner rover, Marie Curie, exists at JPL and can be used for field-testing of the generated sequences. The majority of this work done so far focused on creating a rover model using the ASPEN planning system. The Sojourner planning model was built to a level at which all flight rules and constraints could be implemented. The resources include the three cameras, Alpha Proton X-Ray Spectrometer (APXS), APXS deploy motor, drive motors, solar array, battery, RAM usage, and EEPROM usage. There are 27 different state variables used to track the status of various devices, modes, and parameters. Some of these parameters map directly onto rover internal parameters and others are related to the ASPEN specific model. We are not modeling all rover internal parameters because many are not required for constraint checking. We have defined 162 activities of which 63 decompose directly into low-level rover commands.

There are several constraints that affect overall operations of the Sojourner rover. These include:

- Earth-Mars one-way communications time delay (5-20 minutes)
- Limited communications bandwidth (generally < 10 Mbits downlink per sol[1] available to rover)
- Limited communications opportunities (1 command uplink, 2 telemetry downlinks per sol)

The power system is the single most important resource for the Sojourner Rover. This system consists of a .22 square meter solar array and 9 LiSOCL batteries. The batteries on Sojourner are primarily used during the night for APXS data collection. They are primary batteries and therefore modeled as non-renewable depletable resources. The solar array is the primary power source used

[1] A Sol is a Martian day, equivalent to about 24 hours and 39 minutes

during the day. The predicted available solar power profile throughout the Mars day must be input before planning begins. Using a daily model is required due to changing solar array power available as a result of degradation from dust accumulation and seasonal solar irradiation variability. The angle of the solar array, which depends on the terrain, will also affect the availability of solar energy. Solar array angle estimates could be generated by RCW for input into ASPEN.

A typical Mars day might involve a subset of the following activities:

- Complete an APXS data collection that was carried out during the prior night
- Capture a rear image of the APXS site
- Traverse to an appropriate site and perform a series of soil mechanics experiments
- Traverse to a designated rock or soil location
- Place the APXS sensor head
- Capture end-of-day operations images with its forward cameras
- Begin APXS data collection (usually occurs overnight while the rover is shutdown)
- Shut down for the night

Each of these activities can be input into ASPEN as a goal for that Mars day planning horizon. The format of the input goals is RML or Rover Modeling Language. RML is an application of Extensible Markup Language (XML) designed specifically for rover operations. Both RCW and WITS use RML to communicate with ASPEN. We chose to base our data language on XML for several reasons. First, XML is an emerging data representation standard with widespread support from both proprietary-software and free-software organizations. Because XML is free and open-source, there is a wide community of users supporting development of tools and parsers that make XML easier to use. Second, XML files tend to be naturally modular, creating flexibility for adding mission-specific data. Third, XML is well suited for creating HTML formatted uplink and downlink reports, saving hours of labor.

The exact position of the rover after a traverse activity is subject to dead reckoning error. The timing of traverse activities is also non-determinant. Because of the inherent problems of coordinating activities between the event-based rover and time-based lander, wait commands are used to synchronize activities. When the lander is imaging the rover after a traverse, a wait command is used to ensure the rover will remain stationary at its destination until the lander completes imaging. Because the rover executes commands serially, this ensures that another command will not start execution before the previous command has completed. All rover traverse goals are generated using the RCW. (ASPEN is not designed to perform rover motion planning.) The RCW operator can fly a 3-D rover icon through the stereoscopic display of the Martian terrain. By inspecting the stereo scene, as well as placing the rover icon in various positions within the scene, the operator can assess the trafficability of the terrain. By placing the icon in the appropriate position and orientation directly over the stereo image of the actual rover on the surface, the rover's location and heading are automatically computed. This position information is output to ASPEN to set the rover end position state. The rover driver specifies the rover's destinations by designating a series of waypoints in the scene, generating waypoint traverse commands.

Rover data storage is a scarce resource that must be tracked within the ASPEN model. The largest consumer of data storage is the camera image activity. This activity can fill the on-board data storage if a telemetry session with the lander is not available during the data collection. ASPEN will keep track of the data storage resource to ensure that all data is downlinked before the buffer is completely full.

Initial testing on the Sojourner ASPEN model with a representative set of 136 activities produced a conflict free plan in about 9 seconds. This testing was completed on a Sun Ultra-2 workstation. These relatively quick plan cycles would allow a rover operations team to perform "what-if" analysis on different daily plans. Our goal is that this quick planning capability will be used to generate commands more frequently than once-per-day, if communications opportunities permit.

The next level of testing involved generating plans for two typical Sojourner rover days on Mars. These plans were compared with the manually generated sequences that were run during the Sojourner mission. The command sequences were very similar. The results are summarized in Table 1. Both days produced results very quickly. However, it was a lengthy process (about 10 work weeks) to produce a model that contained constraints and flight rules from a mission not designed for automated planning. Many of the commands were built into macros, which were basically mini-sequences. There was not enough flexibility to utilize all the capabilities of ASPEN in building these plans. If the operations of a mission are designed with an automated planning system in mind, the model building time could be reduced significantly. Once the model is built, valid sequences can be produced very quickly.

	Number of Activities	Planning Time
Sol 18	197	41 seconds
Sol 28	110	6 seconds

Table 1 - Test Results

Eventually we would like to add performance metrics to the planner model to optimize the generated plans. This will enable automated "what-if" analysis to generate plans that maximize science and engineering value.

Future Work: Mars Exploration Rover and Beyond

The goal of this automated planning work is a deployment on a future planetary rover such as the Mars Exploration Rover (MER) mission. (See Figure 3.) Two rovers are planned for launch from Cape Canaveral, Florida, during June 2003 for an early 2004 arrival. The rovers will be identical to each other, but will land at different regions of Mars. Each rover will carry a sophisticated set of instruments, the Athena payload, that will allow it to search for evidence of liquid water that may have been present in the planet's past. Each rover has a mass of nearly 150 kilograms (about 300 pounds) and has a range of up to 100 meters (about 110 yards) per sol, or Martian day.

The Athena payload consists of the Pancam Mast Assembly (PMA), which includes a high-resolution stereo panoramic multi- spectral imaging system (Pancam), and Mini-TES, an emission spectrometer operating in the 5 to 29 micrometer spectral window. Mini-TES is also designed to be a point spectrometer that gathers thermal data as individual spectra or as arrays for key targets identified using Pancam data.

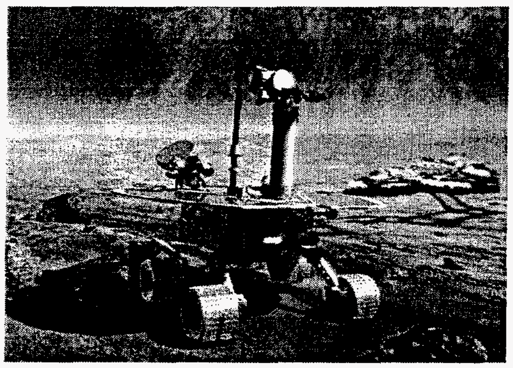

Figure 3 - Mars Exploration Rover

Three additional instruments are to be placed on the end of the Instrument Deployment Device (IDD). The IDD is a deployable arm/instrument package that will perform in-situ analyses of rocks and soils. Instruments on the IDD are the Alpha Particle X-ray Spectrometer (APXS), the Mössbauer Spectrometer, and the Microscopic Imager. Use of all three instruments provides detailed elemental, mineralogical, and textural characterization of rock and soil targets.

The final capability included in the Athena payload is rock abrasion tool, or "RAT," which will be used to expose fresh rock surfaces for study. Additional instruments on the MER rovers will include Navcam stereo imaging systems on the PMA for path planning, and body-mounted Hazcams that image the near terrain to the front and rear of the rover for hazard detection and arm deployment planning.

Each MER rover is designed to conduct traverse science, mast-based remote sensing and in-situ analyses, over a distance of approximately 600 meters during the nominal operational period of 90 sols, but could continue longer, depending on the health of the vehicles. Due to the communication time delays between Earth and Mars, the rover must perform its traverses autonomously, with human operator input generally limited to designation of traverse waypoints and high level commands specifying experiment execution once per sol.

The landed portion of the Mars Exploration Rover mission features a design dramatically different from Mars Pathfinder's. Where Pathfinder had scientific instruments on both the lander and the Sojourner rover, these larger rovers will carry their instruments with them. In addition, these exploration rovers will be able to travel almost as far in one Martian day as the Sojourner rover did over its entire lifetime.

MER has similar operations constraints as previous JPL rovers. Power is the most limited resource, followed by communications bandwidth. The bandwidth is further constrained because there will be two rovers operating simultaneously. Each rover has the ability to communicate directly with Earth through the Deep Space Network, or through the orbiting Mars Odyssey or Mars Global Surveyor using UHF communications. ASPEN is particularly well suited to building schedules that optimize science based on resource constraints such as power and bandwidth.

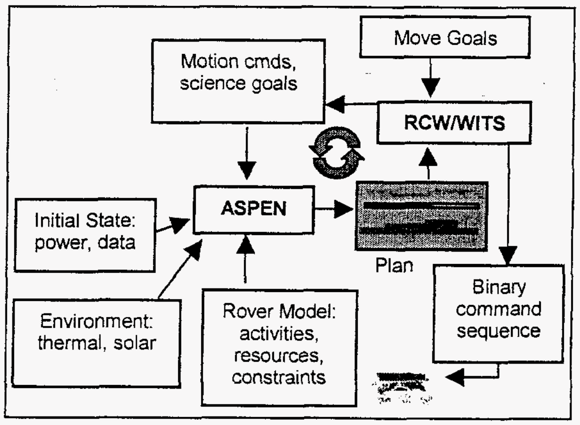

Figure 4 - End-to-End Commanding System

In 2001, we are providing an in-depth validation of the automated command-generation concept using the MER mission. The ASPEN planning and scheduling system will be integrated with the current versions of RCW and WITS. ASPEN will receive RML formatted high-level engineering requests from RCW, and high-level science requests through WITS. ASPEN will then automatically generate validated rover-command sequences that satisfy these requests and provide those RML formatted sequences to RCW. The ASPEN Java-based interface will enable the user to access planned activities and to observe resource and state constraints. The computation intensive aspects of the commanding capability (such as the planner/scheduler, path planner, uncertainty estimation software, vision and image processing software, etc.) will reside on one or more rover workstations based in a central location.

The end-to-end data flow for this system is shown in Figure 4. The interaction between ASPEN and RCW/WITS is an iterative process. RCW will receive high-level motion goals from the user through a 3-dimensional interface utilizing Martian surface imagery. RCW will output detailed traverse commands to ASPEN for inclusion into the schedule. ASPEN will merge these motion commands with high-level science goals from WITS to produce an intermediate level plan. The plan will be output to RCW to update motion commands as necessary. Science goals can be updated through the ASPEN interface or additional high-level science goals can be input through WITS. This process will continue until an acceptable plan is generated. Finally a time ordered list of commands is output for sequence generation.

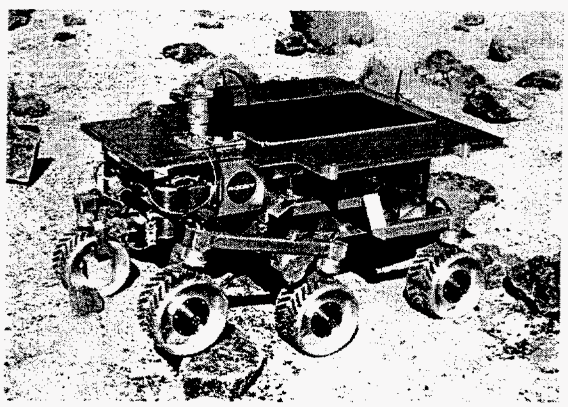

Figure 5 - Rocky 8 Rover

Work is continuing on creating a high-fidelity MER planning model. The automated planning system may be used for goal-based operations during field-testing of MER prior to launch in 2003. The goal of this work is to perform shadow testing in parallel with MER operations to evaluate the effectiveness of automated planning. In addition, we are formulating plans for using this architecture in field-testing of the Rocky-8 rover starting in Fall 2001. (See Figure 5.) These tests would likely be performed initially in the JPL Mars Yard, followed by demonstrations in desert sites in California. The Rocky-8 rover is similar to the rover NASA plans for launch in 2007. The experiences learned from field-testing an automated planner with Rocky-8 will lead to a more robust planning system for the 2007 mission.

A summary of the ground-based planning work is contained in Table 2.

Rover/ Mission	Status of Automated Commanding
Rocky-7	Field tested in 1998 with limited set of goals using WITS interface
Sojourner/ Marie Curie	Fully developed model of rover, flight rules, constraints. Compared with Sojourner surface operations for 2 sample days of operations
MER	Model being built for possible shadow mode testing during field tests and Mars operations using WITS, RCW
Rocky-8	Model will be built Fall 2001 for field testing in early 2002 using Rocky-8 & WITS
2007 Rover	Ground-based automated planning used for operations

Table 2 – Summary of Planning Work

Onboard Rover Planning

In addition to the ground-based planning previously described, we are developing a dynamic, onboard planning system for rover sequence generation. The CASPER (Continuous Activity Scheduling, Planning, Execution and Re-planning) system (Chien et al., 1999; Chien et al., 2000), is a dynamic extension to ASPEN, which can not only generate rover command sequences but can also dynamically modify those sequences in response to changing operating context. If orbital or descent imagery is available, CASPER interacts with a path planner to estimate traversal lengths and to determine intermediate waypoints that are needed to navigate around known obstacles.

Once a plan has been generated it is continuously updated during plan execution to correlate with sensor and other feedback from the environment. In this way, the planner is highly responsive to unexpected changes, such as a fortuitous event or equipment failure, and can quickly modify the plan as needed. For example, if the rover wheel slippage has caused the position estimate uncertainty to grow too large, the planner can immediately command the rover to stop and perform localization earlier than originally scheduled. Or, if a particular traversal has used more battery power than expected, the planner may need to discard one of the remaining science goals. CASPER has been integrated with control software from the JPL Rocky 7 rover (Volpe et al., 2001, Volpe et al., 2000) and is currently being tested on Rocky 7 in the JPL Mars Yard.

Conclusions

Current approaches to rover-sequence generation and validation are largely manual, resulting in an expensive, labor, and knowledge intensive process. This is an inefficient use of scarce science-PI and key engineering staff resources. Automation as targeted by this tool will automatically generate a constraint and flight rule checked time ordered list of commands and provides resource analysis options to enable users to perform more informative and fast trade-off analyses. Initial tests have shown planning times on the order of seconds rather than hours. Additionally, this technology will coordinate sequence development between science and engineering teams and would thus speed up the consensus process.

Enabling goal-driven commanding of planetary rovers by engineering and science personnel greatly reduces the workforce requirements for highly skilled rover engineering personnel. The reduction in team size in turn reduces mission operations costs. In addition, goal-driven commanding permits a faster response to changes in rover state (e.g., faults) or science discoveries by removing the time consuming manual sequence validation process, allowing "what-if" analyses, and thus reducing overall cycle times.

Acknowledgement

The TMOD Technology Program partly funded the research described in this paper. The Mars Surveyor Operations Office, the Athena Rover Office, and the Athena Precursor Experiment funded the RCW development work. All work was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Bibliography

P. Backes, K. Tso, and G. Tharp. "Mars Pathfinder mission Internet-based operations using WITS. Proceedings IEEE International Conference on Robotics and Automation," pages 284-291, Leuven, Belgium, May 1998.

P. Backes, G. Rabideau, K. Tso, S. Chien, "Automated Planning and Scheduling for Planetary Rover Distributed Operations," Proceedings of the IEEE Conference on Robotics and Automation (ICRA), Detroit, Michigan, May 1999.

J. Bresina, K. Golden, D. Smith, and R. Washington, Increasaed Flexibility and Robustness of Mars Rovers, Proceedings of the 5th International Symposium on AI, Robotics, and Automation in Space, Noordwijk, The Netherlands, June 1999.

S. Chien, R. Knight, A. Stechert, R. Sherwood, G. Rabideau, "Integrated Planning and Execution for Autonomous Spacecraft", Proceedings of the IEEE Aerospace Conference (IAC), Aspen, CO, March 1999.

S. Chien, R. Knight, A. Stechert, R. Sherwood, and G. Rabideau, "Using Iterative Repair to Improve Re-sponsiveness of Planning and Scheduling," Proceedings of the 5th Intl. Conference on Artificial Intell-igence Planning and Scheduling, Breckenridge, CO, April 2000.

S. Chien, G. Rabideau, R. Knight, R. Sherwood, B. Engelhardt, D. Mutz, T. Estlin, B. Smith, F. Fisher, T. Barrett, G. Stebbins, D. Tran , "ASPEN - Automating Space Mission Operations using Automated Planning and Scheduling," SpaceOps, Toulouse, France, June 2000.

A. Mishkin, "Field Testing on Mars: Experience Operating the Pathfinder Microrover at Ares Vallis," presentation at Field Robotics: Theory and Practice workshop, May 16 1998, at the 1998 IEEE International Conference on Robotics and Automation, Leuven, Belgium.

A. Mishkin, J. Morrison, T. Nguyen, H. Stone, B. Cooper, B. Wilcox, "Experiences with Operations and Autonomy of the Mars Pathfinder Microrover," proceedings of the 1998 IEEE Aerospace Conference, March 21-28 1998, Snowmass at Aspen, Colorado.

G. Rabideau, R. Knight, S. Chien, A. Fukunaga, A. Govindjee, "Iterative Repair Planning for Spacecraft Operations in the ASPEN System," International Symposium on Artificial Intelligence Robotics and Automation in Space (ISAIRAS), Noordwijk, The Netherlands, June 1999.

R. Volpe, T. Estlin, S. Laubach, C. Olson, and J. Balaram, "Enhanced Mars Rover Navigation Techniques" To appear in the Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, CA, April 2000.

R. Volpe, I. Nesnas, T. Estlin, D. Mutz, R. Petras, H. Das, "The CLARAty Architecture for Robotic Autonomy." Proceedings of the 2001 IEEE Aerospace Conference, Big Sky, Montana, March 10-17, 2001.

Zweben, M., Daun, B., Davis, E., and Deale, M., "Scheduling and Rescheduling with Iterative Repair," Intelligent Scheduling, Morgan Kaufmann, San Francisco, 1994, pp. 241-256.